

This Page Is Inserted by IFW Operations
and is not a part of the Official Record

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

IMAGES ARE BEST AVAILABLE COPY.

**As rescanning documents *will not* correct images,
please do not report the images to the
Image Problems Mailbox.**

THIS PAGE BLANK (USPTO)



19 BUNDESREPUBLIK
DEUTSCHLAND



DEUTSCHES
PATENT- UND
MARKENAMT

12 Off nlegungsschrift
10 DE 198 50 469 A 1

51 Int. Cl.⁷:
G 05 B 15/02
G 08 C 15/00
// G 06 F 13/00

21 Aktenzeichen: 198 50 469.1
22 Anmeldetag: 2. 11. 1998
43 Offenlegungstag: 11. 5. 2000

DE 198 50 469 A 1

71 Anmelder:
Siemens AG, 80333 München, DE

72 Erfinder:
Gehring, Hubert, Dipl.-Inform., 90453 Nürnberg,
DE; Körzdörfer, Horst, Dipl.-Inform., 91056
Erlangen, DE; Talanis, Thomas, Dipl.-Inform., 91336
Heroldsbach, DE; Zebisch, Thomas, Dipl.-Inform.,
91056 Erlangen, DE

55 Entgegenhaltungen:

DE 296 02 226 U1
US 55 13 095

HERBST, L. et. al: "Modernste Prozeßvisualisierung
im Kraftwerk Schkopau", In: ABB Technik 1/1997
S. 13-18;
WEHRES, V.: "Wirtschaftliche und technische
Auswirkungen der digitalen Feldtechnik auf die
Prozeßleittechnik" In: atp-Automatisierungs-
technische Praxis 40 (1998) 3, S. 30, 33 34, 37,
39;

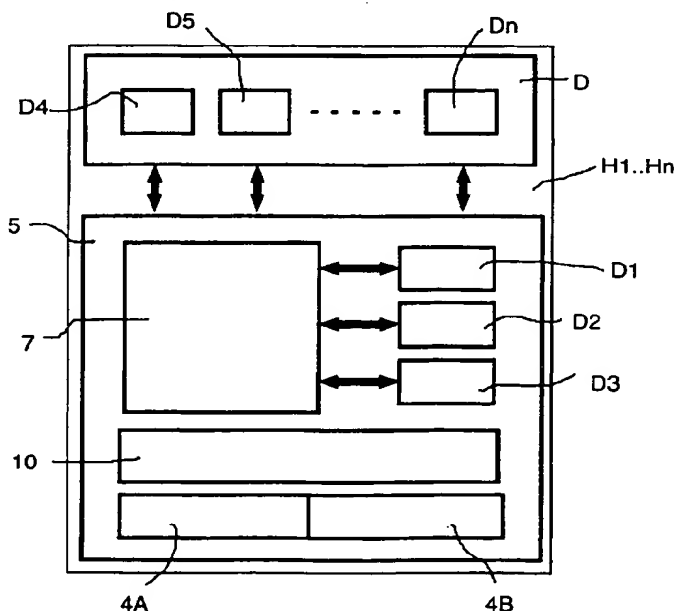
HAPPACHER, M.: "Intelligenz ganz unter" In:
Elektronik 14/1997, S. 82-86;
SCHILLING, C.: "Messen per Netz" In: Elektronik
15/1994, S. 58-64;
ALBERT, W.: "Automatisierung von Chemischen
Forschungs- und Entwicklungsanlagen (F+E-
Anlagen) mit PC-basierenden Prozeßleitsystemen
In: atp-Automatisierungstechnische Praxis 39
(1997), S. 11, 12, 14-16, 18, 19;
KNESEL, W. et. al.: "AST - Das Akbs-Sensor-
Interface für die Automation", Hanser Verlag
1994, Kap. 1;

Die folgenden Angaben sind den vom Anmelder eingereichten Unterlagen entnommen

Prüfungsantrag gem. § 44 PatG ist gestellt

54 Automatisierungssystem und Verfahren zum Zugriff auf die Funktionalität von Hardwarekomponenten

57 Die Erfindung betrifft ein Automatisierungssystem (1) mit über mindestens ein Bussystem (B1...B3) koppelbaren Hardwarekomponenten (H1...Hn), insbesondere Motoren, Sensoren, Aktuatoren etc. Eine möglichst flexible und offene Betriebsweise des Automatisierungssystems kann dadurch erzielt werden, daß jede Hardwarekomponente (H1...Hn) jeweils eine Systemverbindungseinheit (5) mit Funktionsobjekten als Abbild der realen Funktionalität der Hardwarekomponenten (H1...Hn) aufweist, wobei die Funktionsobjekte zum Zugriff auf die Funktionalität der Hardwarekomponenten über das Bussystem vorgesehen sind. Die Funktionsobjekte sind als DCOM-Objekte ausgebildet, so daß eine Nutzung der aus der Bürowelt bekannten Mechanismen zur Ausführung von verteilten Anwendungen möglich wird.



DE 198 50 469 A 1

Die Erfindung betrifft ein Automatisierungssystem mit über mindestens ein Bussystem koppelbaren Hardwarekomponenten, insbesondere Motoren, Sensoren, Aktuatoren, etc.

Die Erfindung betrifft weiter ein Verfahren zum Zugriff auf die Funktionalität von derartigen Hardwarekomponenten.

Ein derartiges Automatisierungssystem besteht beispielsweise aus Motoren, Sensoren, etc., die beispielsweise im Zusammenspiel mit einer oder mehreren übergeordneten Steuerungen einen automatischen Produktionsprozeß bilden. Für die Ansteuerung der Hardwarekomponenten ist dabei in der Regel eine spezielle zwischen Steuerung und Hardwarekomponente zwischengeschaltete Abstraktionshülle erforderlich, die die Vermittlung der Steuerungsbeefehle zwischen Steuerungsvorrichtung und Hardwarekomponente übernimmt. Bei einem Austausch von Hardwarekomponenten bzw. bei einer Ergänzung ist somit neben Änderungen der Hardwarekomponenten selbst in der Regel auch eine Änderung bzw. Anpassung der Steuerungsvorrichtung beispielsweise in Bezug auf Steuerungsprogramme etc. erforderlich.

Der Erfindung liegt die Aufgabe zugrunde, auf einfache Weise eine offene und flexible Betriebsweise des Systems zu erzielen. Insbesondere sollen dabei Hardwarekomponenten ohne Änderungen der Steuerprogramme austauschbar sein.

Diese Aufgabe wird durch ein Automatisierungssystem mit über mindestens ein Bussystem koppelbaren Hardwarekomponenten, insbesondere Motoren, Sensoren, Aktuatoren, etc. gelöst, die jeweils eine Systemverbindungseinheit mit Funktionsobjekten als Abbild der realen Funktionalität der Hardwarekomponenten aufweisen, wobei die Funktionsobjekte zum Zugriff auf die Funktionalität der Hardwarekomponenten über das Bussystem vorgesehen sind.

Diese Aufgabe wird durch ein Verfahren zum Zugriff auf die Funktionalität von Hardwarekomponenten, insbesondere Motoren, Sensoren, Aktuatoren, etc. eines Automatisierungssystems gelöst, bei dem die Kopplung der Hardwarekomponenten innerhalb des Automatisierungssystems über mindestens ein Bussystem erfolgt, wobei auf die Funktionalität der Hardwarekomponenten über das Bussystem mittels einer in der Hardwarekomponente integrierte Systemverbindungseinheit mit Funktionsobjekten als Abbild der realen Hardwarefunktionen zugegriffen wird.

Die Erfindung geht von der Erkenntnis aus, daß es für eine Realisierung der Hardwarekomponenten als "plug and play"-Module erforderlich ist, direkt in der Hardwarekomponente einen speziellen Baustein vorzusehen, auf dem die Funktionsobjekte als Zugang zur Funktionalität der Hardwarekomponenten ablauffähig sind. Dieser spezielle Baustein ist in Form der Systemverbindungseinheit realisiert. Die Systemverbindungseinheit ist mit einem Bussystem des Automatisierungssystems gekoppelt, so daß Kommunikationsdaten beispielsweise von einem Leitsystem zur Hardwarekomponente wie auch von und zu sämtlichen mit dem Bussystem gekoppelten weiteren Komponenten übertragen werden können. Hierdurch können spezielle zwischen einem Leitsystem und den Hardwarekomponenten bisher erforderliche Vermittlungsobjekte entfallen. Darüber hinaus verringert sich durch die Einsparung derartiger als Hardware oder Software vorliegender Vermittlungsobjekte auch die Fehlerhäufigkeit, da die Systemverbindungseinheit im wesentlichen auf Softwareobjekten beruht. Hierdurch wird auch bei einem Austausch einer Hardwarekomponente sowie auch im Betriebsfall eine aktuelle und dynamische Abfrage zur Konfiguration des Automatisierungssystems mög-

lich. Darüber hinaus besteht über die Systemverbindungseinheit jederzeit eine direkte Verbindung zur Funktionalität der Hardwarekomponenten.

Eine Konfiguration oder ein Zugriff auf die bestehenden Hardwarekomponenten des Automatisierungssystems kann in vorteilhafter Weise derart realisiert werden, daß das Automatisierungssystem ein Leitsystem und/oder ein Bedien- und Beobachtungssystem zum Zugriff auf die Funktionsobjekte aufweist.

Die Nutzung der aus der Bürowelt bekannten Mechanismen zur Ausführung von verteilten Anwendungen kann dadurch erfolgen, daß die Funktionsobjekte als DCOM-Objekte ausgebildet sind. DCOM (Distributed Component Object Model) ist ein Mechanismus, der es ermöglicht, über mehrere Computer in einem Netzwerk verteilte Anwendungen auszuführen. Eine verteilte Anwendung kann dabei aus mehreren Prozessen bestehen, die zusammenarbeiten, um eine Aufgabe auszuführen. Diese Prozesse können dabei auf einem oder mehreren Rechnern oder Systemverbindungseinheiten ausgeführt werden.

Die Einbettung der den Hardwarekomponenten zugeordneten Systemverbindungseinheiten in ihre Umgebung kann dadurch erfolgen, daß die Funktionsobjekte mindestens ein erstes Funktionsobjekt zur Erzeugung einer minimalen Funktionalität einer Hardwarekomponente, mindestens ein zweites Funktionsobjekt zur Verschaltung von Funktionsobjekten und mindestens ein drittes Funktionsobjekt zur Auflistung von in der Systemverarbeitungseinheit und/oder auf entfernten Systemverarbeitungseinheiten und/oder entfernten Rechnern vorhandenen Funktionsobjekten enthalten.

Ein sogenanntes Verknüpfungsobjekt kann dadurch geschaffen werden, daß das zweite Funktionsobjekt zur Beeinflussung von Verbindungen zwischen den auf entfernten Systemverbindungseinheiten und/oder auf entfernten Rechnern vorhandenen Funktionsobjekten vorgesehen ist.

Ein störungsfreier Zugriff auf die Hardwarekomponenten ohne die Notwendigkeit von Synchronisierungsmaßnahmen kann dadurch sichergestellt werden, daß durch das dritte Funktionsobjekt beim Zugriff auf die Systemverbindungseinheit ein dynamisches Abfrageobjekt generiert wird. Die Besonderheit dieses Abfrageobjekts besteht darin, die Summe der Funktionalität des Systems zu enumerieren.

Eine einfache und wirkungsvolle Möglichkeit zur Bereitstellung von Codierungsmitteln wird dadurch erreicht, daß das Automatisierungssystem eine ladbare Tabelle mit Codierungsvorschriften aufweist. Hierdurch wird auch eine sogenannte Marshaling-Funktion der Funktionsobjekte erzielt.

Im folgenden wird die Erfindung anhand der in den Figuren dargestellten Ausführungsbeispiele näher beschrieben und erläutert.

Es zeigen:

Fig. 1 ein Blockschaltbild eines ersten Ausführungsbeispiels eines Automatisierungssystems mit verteilten Objekten,

Fig. 2 ein weiteres Ausführungsbeispiel eines Automatisierungssystems mit mehreren Bussystemen,

Fig. 3 eine schematische Darstellung für den Aufbau einer Systemverbindungseinheit einer Hardwarekomponente und

Fig. 4 ein abstraktes Geräteobjektmodell.

Fig. 1 zeigt ein Blockschaltbild eines Automatisierungssystems **1**. Das Automatisierungssystem **1** besteht aus einem Leitsystem **3**, welches über ein Bussystem **B1** mit einer Vielzahl von Hardwarekomponenten **H1..Hn** gekoppelt ist. Mit dem Bussystem **B1** ist weiterhin ein Bedien- und Beobachtungssystem **6** gekoppelt. Jede Hardwarekomponente **H1..Hn** weist eine Systemverbindungseinheit **5** auf, die jeweils mit dem Bussystem **B1** gekoppelt ist.

Zentraler Bestandteil des in Fig. 1 dargestellten Automatisierungssystems 1 ist die Systemverbindungseinheit 5, die der Kommunikation der Hardwarekomponenten H1..Hn innerhalb des Automatisierungssystems 1 sowie zur Abbildung der jeweiligen Funktionalität der Hardwarekomponenten H1..Hn, d. h. zur Erzeugung von Repräsentanten hierfür dient. Hierzu weist die Systemverbindungseinheit 5 ein minimales gemeinsames Interface auf, welches eine für alle Hardwarekomponenten H1..Hn vorgegebene Identifizierung enthält und hierdurch von einem externen Kommunikationspartner, beispielsweise vom Leitsystem 3 auch ohne Kenntnis über das eigentliche Gerät, d. h. über die eigentliche Hardwarekomponente H1..Hn angesprochen werden kann. Durch die Systemverbindungseinheit 5 kann vom jeweiligen Ansprechpartner der Hardwarekomponente H1..Hn nach Aufnahme einer Kommunikation weiteres Wissen über die jeweilige Hardwarekomponente H1..Hn erlangt werden. Mit Hilfe der Systemverbindungseinheit 5 wird es somit möglich, Hardwarekomponenten H1..Hn ohne Änderung der bestehenden Strukturen des Automatisierungssystems 1 zu ersetzen, zu ergänzen etc. Darüber hinaus entfallen aufgrund der direkten Kommunikation eines übergeordneten Leitsystems mit der in der Hardwarekomponente H1..Hn angeordneten Systemverbindungseinheit 5 ansonsten erforderliche dazwischengeschaltete Abstraktionshüllen. Insgesamt führt das in Fig. 1 dargestellte Grundprinzip von Hardwarekomponenten H1..Hn mit integrierter Systemverbindungseinheit 5 dazu, daß das Automatisierungssystem 1 offen und flexibel betrieben werden kann. Dies wird im folgenden auch mit dem Schlagwort "offene/verteilte Automatisierung" bezeichnet werden.

Fig. 2 zeigt ein weiteres Ausführungsbeispiel eines Automatisierungssystems mit mehreren Bussystemen B1, B2, B3. Mit den ersten Bussystem B1 ist eine Hardwarekomponente H1 gekoppelt, die einen Motor M1 wie eine zugehörige Signalverarbeitungseinheit 5 enthält. Die Hardwarekomponente H1 ist darüber hinaus mit dem zweiten Bussystem B2 gekoppelt, welches mit einem Leitsystem 3 verbunden ist. Das Leitsystem 3 ist darüber hinaus mit dem dritten Bussystem B3 koppelbar. An das dritte Bussystem B3 ist ein Bedien- und Beobachtungssystem 6 sowie eine speicherprogrammierbare Steuerung 8 verbunden.

Die Hardwarekomponente H1, genauer gesagt die Systemverbindungseinheit 5 der Hardwarekomponente H1 mit zugehörigem Signalverarbeitungsmittel ist in der Lage über das erste B1 und das zweite B2 Bussystem Kommunikationsdaten zu übertragen bzw. zu empfangen. Darüber hinaus bildet die Systemverbindungseinheit 5 auch eine Schnittstelle für einen Netzübergang zwischen dem ersten Bussystem B1 und dem zweiten Bussystem B2, beispielsweise für Kommunikationsdaten, die von einer mit dem Bussystem B1 gekoppelten Hardwarekomponente H2 an das Leitsystem 3 übertragen werden sollen. Hierzu weist die Systemverbindungseinheit 5 einen Speicher 9 zur Speicherung von für einen Netzübergang zwischen dem Bussystem B1 und dem Bussystem B2 erforderlichen Protokollen auf. So ist beispielsweise auf einfache Weise ein Netzübergang zwischen einem sogenannten Ethernet und einem sogenannten Profibus möglich. Die Hardwarekomponente H1 weist bei dem in Fig. 2 dargestellten Ausführungsbeispiel neben dem Vorteil einer offenen und verteilten Kommunikation zwischen der Hardwarekomponente H1 und beispielsweise dem Leitsystem 3 auch den weiteren Vorteil einer Schnittstellenanpassung für weitere Hardwarekomponenten auf.

Fig. 3 zeigt eine schematische Darstellung des Aufbaus einer Systemverbindungseinheit 5 einer Hardwarekomponente H1..Hn, wie sie beispielsweise im Zusammenhang mit dem in Fig. 1 und Fig. 2 dargestellten Ausführungsbeispielen

zum Einsatz kommt. Die Signalverbindungseinheit 5 enthält Funktionsobjekte D1, D2, D3 sowie Betriebssystemkomponenten 4A, 4B, ein Laufzeitsystem 7 (Runtime-System) sowie eine Protokollverarbeitungseinheit 10 (Profibus, UDP/IP, RPC). Die Hardwarekomponenten H1..Hn weist darüber hinaus weitere Datenobjekte D4..Dn auf, die als Datenobjekte D eines Anwenders eine zusätzliche Anwenderfunktionalität kennzeichnen.

Die Funktionsobjekte D1..D3 sind beispielsweise als sogenannte DCOM-Objekte (Distributed Component Object Model) bzw. als sogenannte OLE-Objekte (Object Linking and Embedding) ausgebildet. Das erste Funktionsobjekt D1 dient der Erzeugung einer minimalen Funktionalität einer Hardwarekomponente. Das zweite Funktionsobjekt D2 dient zur Verschaltung von Objekten, während das dritte Funktionsobjekt D3 zur Sammlung von in der Systemverbindungseinheit und/oder auf entfernten Systemverarbeitungseinheiten und/oder entfernten Rechnern vorhandenen Objekten vorgesehen ist. Das zweite Funktionsobjekt D2 benutzt das Laufzeitsystem 7, um Verbindungen herzustellen. Das Laufzeitsystem 7 baut auf dem Remote Procedure Call RPC-Protokoll auf und ist eine dem RPC-Protokoll überlagerte Verwaltungseinheit, die eine Sicht auf lokale oder entfernte Objekte bzw. Interfaces mit Hilfe der RPC-Protokolle herstellt. Das besondere der Laufzeitschnittstelle 7 ist es, daß die Schnittstelle asynchron ausgebildet ist, d. h. nach Rückkehr der entsprechenden Funktion liegen die Ergebnisse, die von einem entfernten System zu liefern sind, noch nicht vor. Der Aufrufer, d. h. der Auslöser des Remote Procedure Calls muß daher die Adresse einer Variablen an das Laufzeitsystem 7 übergeben. Das Laufzeitsystem 7 verändert beim Eintreffen der Ergebnisse oder einem vorgebbaren Time-out den Wert einer bestimmten Quittungskomponente. Hierdurch wird dem Aufrufer mitgeteilt, ob die Operation erfolgreich abgeschlossen wurde oder fehlschlug. Der Vorteil der Asynchronität der Funktionsschnittstelle des Laufzeitsystems des zweiten Datenobjekts D2 ist es, daß beispielsweise bei einer zyklischen speicherprogrammierbaren Steuerung mit einem strengen Zyklus der vorgegebene Zyklus eingehalten werden kann. Hierzu ruft das Anwenderprogramm in einem beliebigen Zyklus die Methode eines lokalen bzw. entfernten Objekts, während das Ergebnis des Aufrufs beispielsweise per Flag im nächsten Zyklus abgefragt wird.

Die Registrierung von Hardwarekomponenten H1..Hn erfolgt in der Weise, daß mit Hilfe einer bestimmten Funktion, d. h. eines Befehls ein Interface sichtbar im Laufzeitsystem 7 anmeldet. Das registrierte Interface ist danach für sogenannte "Clients" zugänglich. Künftige Aufrufe von derartigen Clients werden vom Laufzeitsystem 7 an die vorher übergebene Funktion weitergeleitet.

Die Systemverbindungseinheit 5 enthält Umsetzungsmittel zur Umsetzung zwischen Rechnerinformationsdaten und Daten auf einem Kommunikationsmedium. Dabei liegt die Verantwortung für eine derartige Umsetzung, d. h. für das sogenannte Marshaling beim jeweiligen Datenobjekt D1..Dn selbst. Die Datenobjekte D1..Dn müßten vor dem Aufruf einer entfernten Methode entsprechend der Parametersignatur der jeweiligen Methode einen Puffer mit Werten füllen, der den Codierungsvorschriften von beispielsweise RPC bzw. DCOM gerecht wird. Entsprechendes gilt für die Rückgabe der Out-Parameter, die Entgegennahme der In-Parameter und die Entgegennahme von Out-Parametern.

Das DCOM-Runtime-System 7 stellt Funktionen für das Marshaling einzelner Werte zur Verfügung.

Das DCOM-Runtime-System 7 stellt die Verbindungen zwischen Objekten auf unterschiedlichen Rechnern her. Für diese Aufgabe wird zum einen eine Liste von Interfaces ver-

waltet, die von lokalen Clients auf entfernten Rechnern beansprucht werden, sogenannte importierte Interfaces. Zum anderen werden Listen mit Interfaces verwaltet, die lokale Server für entfernte Clients zur Verfügung stellen, sogenannte exportierte Interfaces.

Fig. 4 zeigt ein abstraktes Geräteobjektmodell M für alle an einem offenen verteilten Automatisierungssystem teilnehmenden Komponenten OVA-Komponenten 13 (OVA-Device = Offene Verteilte Automatisierung). Das Geräteobjektmodell M besteht im wesentlichen aus einem allgemeinen Gerät A, welches als Systemdefinition die über alle Geräte gleiche Funktionalität eines sogenannten OVA-Geräts (Offene Verteilte Automatisierung) beschreibt. Das allgemeine Gerät A wird durch die drei Funktionsobjekte D1, D2, D3 realisiert. Das allgemeine Gerät A enthält weiter ein erweitertes Gerät E beschreibt alle Dienste des Geräts, die eine bestimmte Geräteklasse, beispielsweise einen bestimmten Gerätetyp von speicherprogrammierbaren Steuerungen zusätzlich zum allgemeinen Objektmodell A anbieten. Es wird über ein Objekt E1 (Extended Device) beschrieben. Das Geräteobjektmodell M enthält weiter einen Bereich 14 für die einer OVA-Komponente 13 zugeordnete Anwenderfunktionalität. Die Anwenderfunktionalität der Automatisierungsobjekte 13 kann dabei über ein weiteres Objekt 12, beispielsweise über ein sogenanntes ACAO-Interfaces (Active Control Application Object) beschrieben werden. Über das ACAO-Interface bestehen Interfaces zu den Datenobjekten D4..Dn.

Das Objekt D1 (Common Device) stellt die allgemeine minimale Funktionalität eines Geräts zur Verfügung. Es hat eine für alle Geräte gleiche wohlbekannte Identifizierung und kann somit einen externen Kommunikationspartner auch ohne Kenntnis über das eigentliche Gerät angesprochen werden. Das Objekt D1 ist sozusagen Anker für alle weiteren Objekte des Geräts, d. h. die auf dem Gerät lebenden Objekte können über das Objekt D1 bestimmt werden. Das Objekt D2 (ACCO = Active Control Connection Object) realisiert die Verschaltung von Automatisierungsobjekten, während das Objekt D3 (RT Automation Object) Verweise auf die Automatisierungsobjekte des Geräts enthält. Die besondere Funktion des Funktionsobjekts D3 besteht darin, die Summe der Funktionalität des Systems zu enumerieren, d. h. zu erfragen. Das erweiterte Gerät E beschreibt alle Dienste des Geräts, die eine bestimmte Geräteklasse, beispielsweise einen bestimmten Gerätetyp von speicherprogrammierbaren Steuerungen zusätzlich zum allgemeinen Objektmodell A anbieten. Es wird über das Objekt Extended Device E1 beschrieben.

Das Geräteobjektmodell ist gemäß Fig. 4 somit in einen Teil fester Funktionalität A sowie in einen Teil E mit variabler Funktionalität unterteilt. Die Funktionalität der Funktionsobjekte D2, D3 wird über das erste Funktionsobjekt D1 zugänglich gemacht, während vom Funktionsobjekt D3 die weiteren Funktionsobjekte D4, ... Dn erfragbar und zugänglich sind. Die Funktionalität der Automatisierungsobjekte kann dabei über das ACAO-Interface 12 (Active Control Application Object) beschrieben werden.

Zusammenfassend betrifft die Erfindung somit ein Automatisierungssystem 1 mit über mindestens ein Bussystem B1..B3 koppelbaren Hardwarekomponenten H1..Hn, insbesondere Motoren, Sensoren, Aktuatoren, etc. Eine möglichst flexible und offene Betriebsweise des Automatisierungssystems kann dadurch erzielt werden, die jeweils eine Systemverbindungseinheit mit Funktionsobjekten als Abbild der realen Funktionalität der Hardwarekomponenten aufweisen, wobei die Funktionsobjekte zum Zugriff auf die Funktionalität der Hardwarekomponenten über das Bussystem vorgesehen sind. Die Funktionsobjekte sind als DCOM-Objekte

ausgebildet, so daß eine Nutzung der aus der Bürowelt bekannten Mechanismen zur Ausführung von verteilten Anwendungen möglich wird.

Patentansprüche

1. Automatisierungssystem (1) mit über mindestens ein Bussystem (B1..B3) koppelbaren Hardwarekomponenten (H1..Hn), insbesondere Motoren, Sensoren, Aktuatoren, etc., die jeweils eine Systemverbindungseinheit (5) mit Funktionsobjekten (D1..Dn) als Abbild der realen Funktionalität der Hardwarekomponenten (H1..Hn) aufweisen, wobei die Funktionsobjekte (D1..Dn) zum Zugriff auf die Funktionalität der Hardwarekomponenten (H1..Hn) über das Bussystem (B1..B3) vorgesehen sind.
2. Automatisierungssystem nach Anspruch 1, dadurch gekennzeichnet, daß das Automatisierungssystem ein Leitsystem (3) und/oder eine Bedien- und Beobachtungssystem zum Zugriff auf die Funktionsobjekte (D1..Dn) aufweist.
3. Automatisierungssystem nach einem der Ansprüche 1 oder 2, dadurch gekennzeichnet, daß die Funktionsobjekte (D1..Dn) als DCOM-Objekte ausgebildet sind.
4. Automatisierungssystem nach einem der Ansprüche 1 bis 3, dadurch gekennzeichnet, daß die Funktionsobjekte (D1..D3) mindestens ein erstes Funktionsobjekt (D2) zur Erzeugung einer minimalen Funktionalität einer Hardwarekomponente (H1..Hn), mindestens ein zweites Funktionsobjekt (D2) zur Verschaltung von Funktionsobjekten und mindestens ein drittes Funktionsobjekt (D3) zur Sammlung von in der Systemverarbeitungseinheit (5) und/oder auf entfernten Systemverarbeitungseinheiten und/oder entfernten Rechnern vorhandenen Funktionsobjekten (D1..Dn) aufweisen.
5. Automatisierungssystem nach einem der Ansprüche 1 bis 4, dadurch gekennzeichnet, daß das zweite Funktionsobjekt (D2) zur Beeinflussung von Verbindungen zwischen den auf entfernten Systemverbindungseinheiten und/oder auf entfernten Rechnern vorhandenen Funktionsobjekten vorgesehen ist.
6. Automatisierungssystem nach einem der Ansprüche 1 bis 5, dadurch gekennzeichnet, daß durch das dritte Funktionsobjekt (D3) beim Zugriff auf die Systemverbindungseinheit ein dynamisches Abfrageobjekt generiert wird.
7. Automatisierungssystem nach einem der Ansprüche 1 bis 6, dadurch gekennzeichnet, daß das Automatisierungssystem eine ladbare Tabelle mit Codierungsvorschriften aufweist.
8. Automatisierungssystem nach einem der Ansprüche 1 bis 7, dadurch gekennzeichnet, daß das Automatisierungssystem zum Zugriff auf die Funktionalität der Hardwarekomponenten ein Leitsystem und/oder eine Bedien- und Beobachtungssystem aufweist.
9. Verfahren zum Zugriff auf die Funktionalität von Hardwarekomponenten (H1..Hn), insbesondere Motoren, Sensoren, Aktuatoren, etc. eines Automatisierungssystems (1), bei dem die Kopplung der Hardwarekomponenten (H1..Hn) innerhalb des Automatisierungssystems über mindestens ein Bussystem (B1..B3) erfolgt, wobei auf die Funktionalität der Hardwarekomponenten über das Bussystem mittels einer in der Hardwarekomponente integrierte Systemverbindungseinheit (5) mit Funktionsobjekten (D1..Dn) als Abbild der realen Hardwarefunktionen zugegriffen wird.
10. Verfahren nach Anspruch 9, dadurch gekennzeichnet, daß in der Systemverbindungseinheit (5) über ein

erstes Funktionsobjekt (D1) eine minimalen Funktio-
nalität einer Hardwarekomponente (H1..Hn) erzeugt
wird, daß über ein zweites Funktionsobjekt (D2) Funk-
tionsobjekte (D1..Dn) miteinander verschaltet werden
und daß über ein drittes Funktionsobjekt (D3) die in der 5
Systemverarbeitungseinheit (5) und/oder auf entfern-
ten Systemverarbeitungseinheiten und/oder auf ent-
fernten Rechnern vorhandene Funktionsobjekte
(D1..Dn) aufgelistet werden.

Hierzu 2 Seite(n) Zeichnungen

10

15

20

25

30

35

40

45

50

55

60

65

- Leerseite -

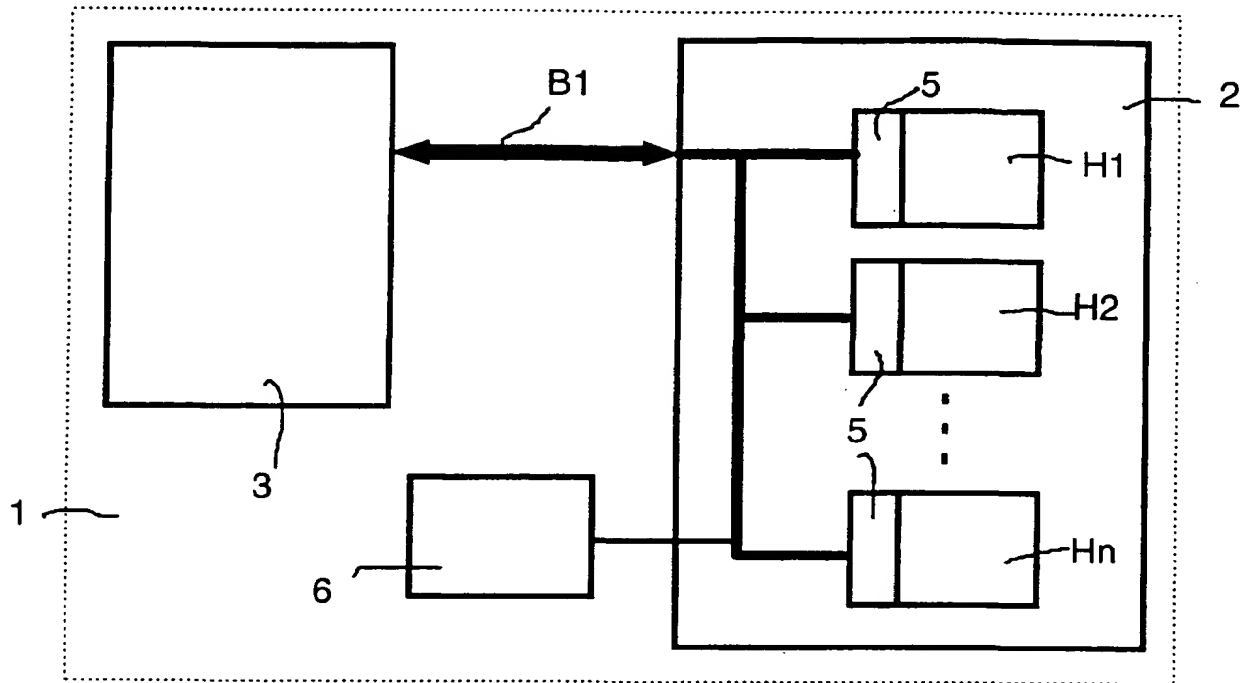


Fig. 1

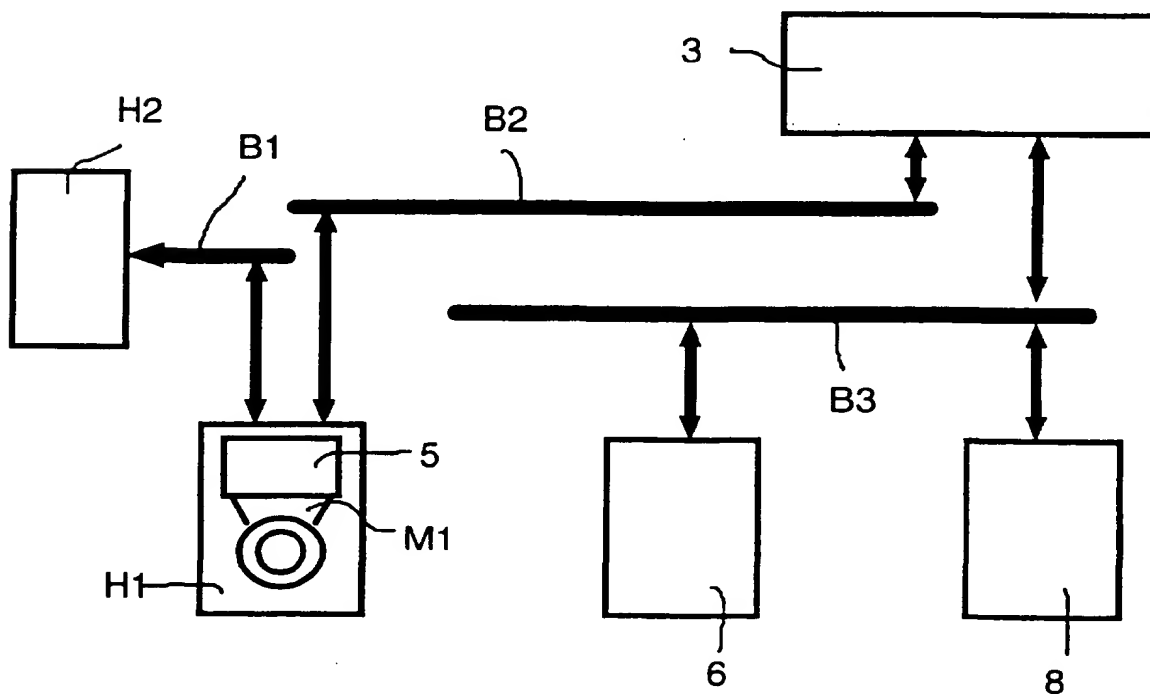


Fig. 2

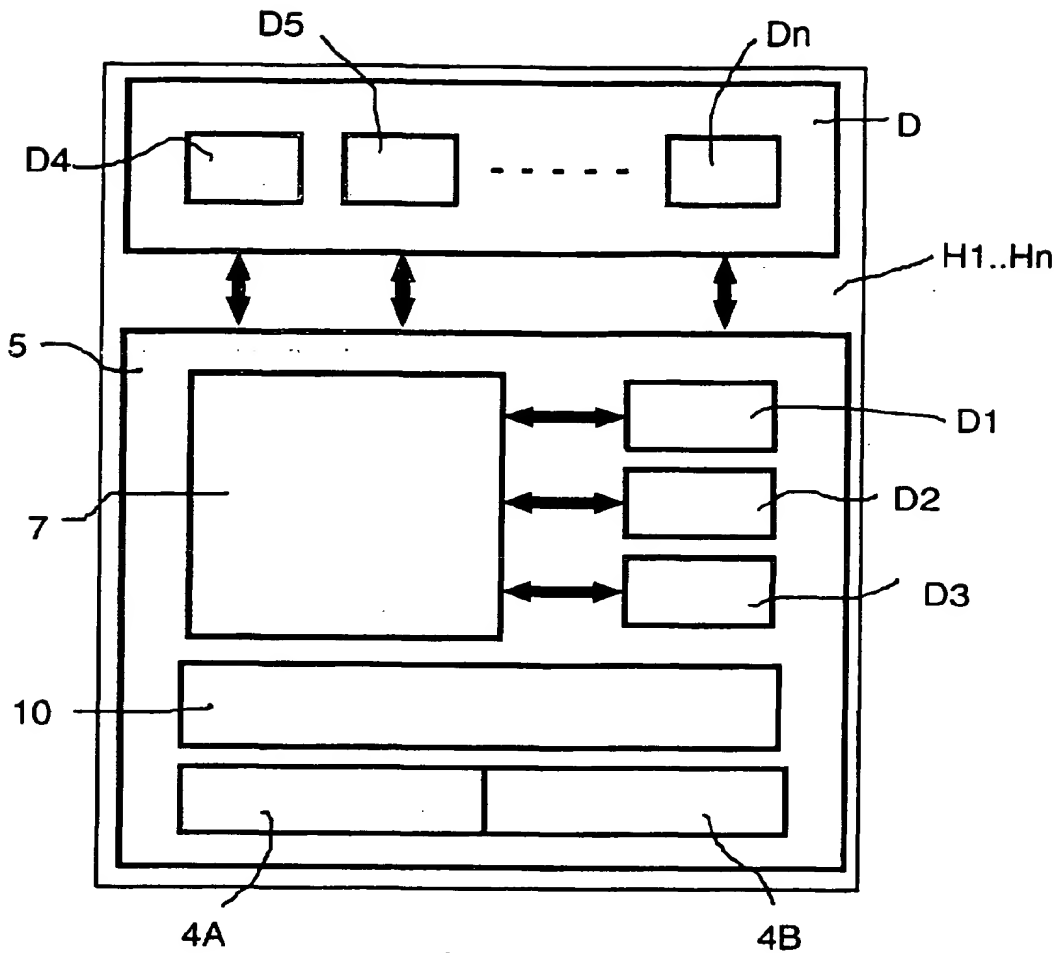


Fig. 3

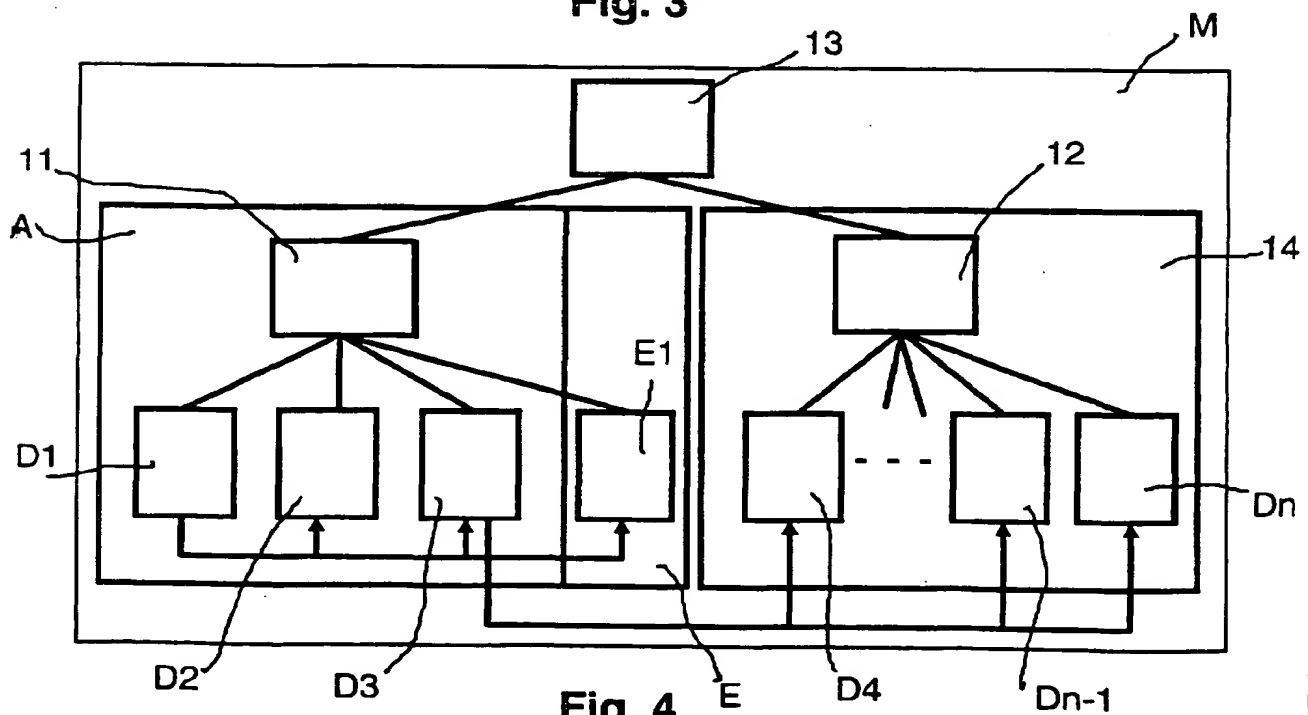


Fig. 4